第4章　アレイ・アンテナを用いて電波の到来方向が分かる

到来方向推定システムの基礎と実装例

Minseok Kim

これまでアレイ・アンテナを用いて所望の信号をうまく合成する方法について説明した．しかし，実はこれ以外にもさまざまなアプリケーションがある．本章では，アレイ・アンテナを用いた電波の到来方向推定法（Direction of Arrival Estimation）について解説し，FPGAを用いた回路実装を紹介する．

（筆者）

1．電波の到来方向推定（DOA Estimation）

最近，携帯電話が目覚ましい勢いで普及しています．しかし，建物の中などには電波の届かない場所が多く存在し，対策としてリピータ（Repeater）と呼ばれる電波中継器を利用している場合が多くあります．ここで，認可を受けていない違法なリピータが使用された場合，これらの放射する不法電波がノイズとなり，携帯電話やそのほかの通信環境を悪化させるとして問題となっています（**図1**）．

このため，不法リピータを検出する装置として，複数の地点において不法電波の到来方向（DOA ：Direction of Arrival）を測定し，その位置が特定できるシステムがすでに開発されています．また，アレイ・アンテナを用いて推定する手法もいろいろと検討されています．

携帯電話による緊急時の通報が増加しています．米国連邦通信委員会（FCC）が緊急通話サービスの向上のため，緊急時の携帯電話利用者の居場所を特定できる機能Enhanced 911（E-911）の制度化を行いました．高精度の位置測定技術が望まれています(1)．やはりこの場合にも，アレイ・アンテナを用いた高精度の到来方向推定が重要な技術として認識されています（**図2**）．

さらに，**図3**のようにセルラ移動通信において，ユーザ端末の到来方向情報を有効に利用することが考えられます．例えば，所望ユーザの方向には送信電力を集中させ，ほか

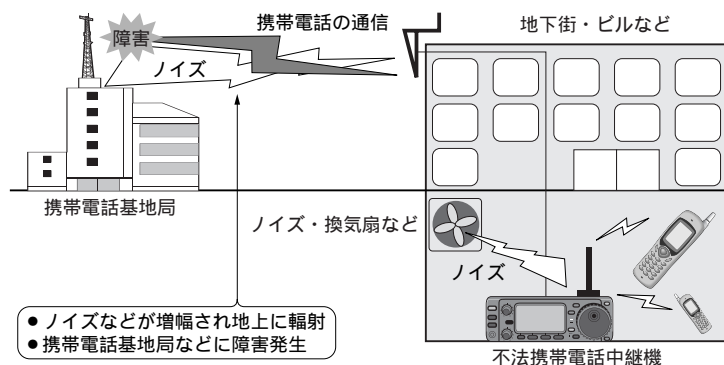



図1　不法中継器の問題

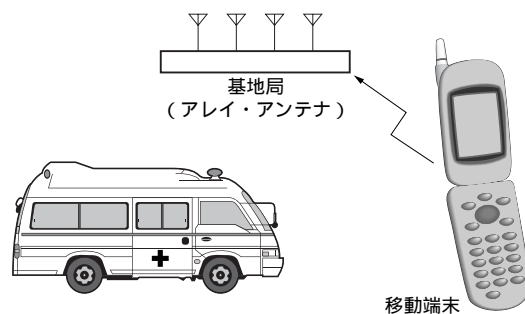



図2　救急システムにおけるアレイ・アンテナの応用

KeyWord　電波の到来方向推定法，フーリエ変換，ビーム・フォーマ法，多重信号分離法，MUSIC，固有値分解

のユーザ方向には指向性のヌルを形成することで干渉を低減できます．搬送波周波数が上下回線で異なる周波数分割複信（FDD：Frequency Division Duplex）において有効な方法として知られています．到来方向情報は上下回線において統計的に等しいことが分かっているので，上り回線（移動端末→基地局）で推定した到来方向情報を下り回線（基地局→移動端末）に用いることができます．ここでは，伝搬路の変化に追いつけるように高速で，かつ，高精度の到来方向推定処理が必要になります．

このように，電波を用いた無線局端末の位置推定は，そのほかにもレーダ・システム，伝搬路同定システムから，近年ではさまざまな無線通信システムやセンサ・ネットワーク，コグニティブ無線まで，その応用が広がりつつあるといえます．

アレイ・アンテナを用いた到来方向推定手法は，フーリエ変換の原理に基づいています．最も基本的な手法のビーム・フォーマ法（Beamformer），線形予測法（Linear Prediction）に基づく手法としてCapon法や最大エントロピ法，さらに，アレイ入力信号の相関行列の固有値展開に基づく手法として多重信号分離法（以下MUSIC：Multiple Signal Classification）やESPRIT（Estimation of Signal Parameters via Rotational Invariance Techni ques）法が考案されました．また，マルチパスの影響を受けやすい環境にはチャネル特性のデータベース化による指紋法（Finger Print）などが提案されています．

これについてもっと詳細な理論が知りたい方は，参考文献[2]を参考にしてください．

本章では，アレイ・アンテナにおける到来方向推定法について，その基礎原理を説明し，現在，最も幅広く利用されている多重信号分離法（MUSIC）について解説します．また，FPGAを用いたハードウェア構成と信号処理部の実装例を簡単に紹介します．

2．到来方向を推定する方法

到来方向推定手法についてまともに説明しようとすると，非常に専門的な話になってしまいます．ここでは，その原理を直感的に捉えられるように，できる限り簡単におさらいをします．

アレイ・アンテナによる到来波の方向推定には，古い歴史があります．指向性ビームによる空間走査を行うビーム・フォーマ法が最も簡単な手法と知られています．ビーム・フォーマ法は，一様励振アレイ・アンテナのメイン・ローブ（主ビーム）を全方向に対して走査し，得られた電力スペクトルから出力電力が大きくなる方向を探索する方法です[2]．

図4のように，重み係数をある角度θにするためには，第2章の説明より以下のようにアレイ・ステアリング・ベクトルに設定することで，メイン・ビームを角度θに向けることができます．

$$\boldsymbol{\omega}(\theta)=\left[1,\exp\left(-j\frac{2\pi}{\lambda}d\sin\theta\right),\cdots \exp\left(-j\frac{2\pi}{\lambda}d(M-1)\sin\theta\right)\right]^{\mathrm{T}} \quad \cdots\cdots(1)$$

ここで，Mはアンテナ素子数，dはアンテナ素子間隔を示します．このとき，アレイ出力は以下のようになります．

$$y(\theta)=\boldsymbol{\omega}^{\mathrm{H}}(\theta)\cdot\boldsymbol{x} \quad \cdots\cdots(2)$$

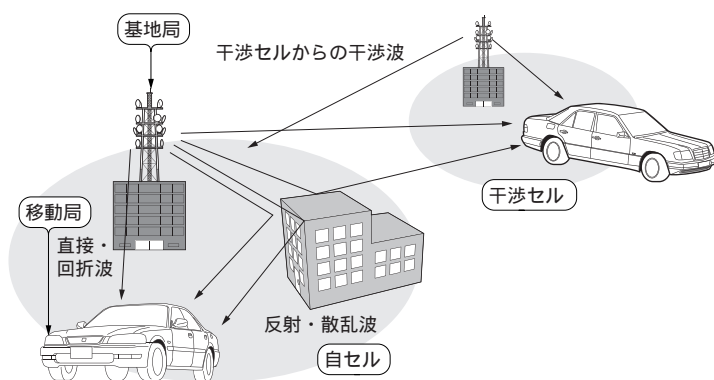
（a）移動通信における伝搬状況

（b）到来方向から所望ユーザに電力を集中させることができる

図3　セルラ移動通信における到来方向推定の利用

ここで，xはアレイ入力信号ベクトル，記号Hは複素共役転置を示します．正規化電力スペクトルは以下の式で得られます．

$$P = \frac{E[y(\theta) \cdot y^*(\theta)]}{\boldsymbol{\omega}(\theta) \cdot \boldsymbol{\omega}^{\mathrm{H}}(\theta)} \quad \cdots\cdots (3)$$

ここで，記号＊は共役を示します．

ビーム・フォーマ法は，空間領域でのフーリエ変換で関係付けられ，FFT（Fast Fourier Transform）を用いて高速処理が可能になります．このような電力検出方法では，理想的なビームはある角度に対するインパルス的な応答を持ちます．現実的には制限されたアレイ・アンテナ素子の数により，メイン・ローブ（メイン・ビーム）の半値幅が広くなります．また，サイド・ローブからも信号を受信してしまうため，一般に角度分解能が悪いことが知られています．しかし，システム的には安定に動作し，最も簡単に実現できることから，現在も幅広く用いられています(2)．

ディジタル制御が自由にできるようになってから，さまざまな手法が考案されてきました．その中で，MUSICやESPRITという部分空間法を用いる超高分解能手法が考案され，現在，最も注目されています(3)，(4)．

（a）リニア・アレイ・ブロック図

（b）8素子アレイ・アンテナに+30°方向から入射されるときの電力スペクトル

図4　ビーム・フォーマ法の原理

3．多重信号分離法（MUSICアルゴリズム：MUltiple SIgnal Classification）

上記のようにビーム・フォーマ法では，メイン・ビームの半値幅によって角度分解能が決まるので，到来波の細かい分離はできません．MUSICアルゴリズムは，アレイ入力データから作成した自己相関行列の固有値展開により，信号部分空間と雑音部分空間に分離できます．それらがお互いに直交するという面白い性質を用いることで，超高分解能の到来信号の推定精度が実現できます．本章では少し専門的な話になりますが，その原理について簡略に説明します．

MUSICアルゴリズムにおける信号処理は**図5**のようになります．アレイ・アンテナに入力された信号は，受信器で周波数変換とA-D変換された複素ディジタル信号になります．各素子での雑音を含んだ受信信号ベクトル$\boldsymbol{X}(t)$から以下のように自己相関行列を作成します．

$$\boldsymbol{R}_{XX} = E\left[\boldsymbol{X}(t) \cdot \boldsymbol{X}^{\mathrm{H}}(t)\right] \quad \cdots\cdots (4)$$

ここで，$E[\cdot]$は期待値を示します．この相関行列は，信号成分とノイズ成分として，以下のように分離して考えられます．信号ベクトル$\boldsymbol{S}(t)$の相関行列を$\boldsymbol{R}_{SS}$，複素白色ガウス・ノイズを$\boldsymbol{N}(t)$とすると，ノイズ相関行列はその分散値と単位行列から$\sigma^2\boldsymbol{I}$となり，$\boldsymbol{R}_{XX}$は以下のように変形できます．

$$\begin{aligned}\boldsymbol{R}_{XX} &= \boldsymbol{A} \cdot E\left[\boldsymbol{S}(t) \cdot \boldsymbol{S}^{\mathrm{H}}(t)\right] \cdot \boldsymbol{A}^{\mathrm{H}} + E\left[\boldsymbol{N}(t) \cdot \boldsymbol{N}^{\mathrm{H}}(t)\right] \\ &= \boldsymbol{A} \cdot \boldsymbol{R}_{SS} \cdot \boldsymbol{A}^{\mathrm{H}} + \sigma^2 \boldsymbol{I}\end{aligned} \quad \cdots\cdots (5)$$

実際に信号とノイズの分離は固有値分解により行われます．一般に相関行列$\boldsymbol{R}_{XX}$は非不定値エルミート行列（Positive Definite Hermitian Matrix）になり,固有値分解によりベクトル空間を次式のように信号部分空間とノイズ部分空間として分離できます．

$$\boldsymbol{R}_{XX} = \boldsymbol{U}_S \cdot \boldsymbol{\Lambda}_S \cdot \boldsymbol{U}_S^H + \boldsymbol{U}_N \cdot \boldsymbol{\Lambda}_N \cdot \boldsymbol{U}_N^H \quad \cdots\cdots (6)$$

ここで$\boldsymbol{U}_S$と$\boldsymbol{U}_N$は，それぞれ信号とノイズの部分空間の固有ベクトルです．$\boldsymbol{\Lambda}_S$は対角成分が実数の固有値｛λ

$_1$, λ_2, …, λ_L}($\lambda_1 \geqq \lambda_2 \geqq \cdots \geqq \lambda_L$)の対角行列で，$\Lambda_N$は対角成分を$\sigma^2$とする対角行列になります．ノイズ部分空間の固有ベクトルは，信号部分空間の固有ベクトルと直交する性質があります．

全方向に対するステアリング・ベクトルとノイズ固有ベクトルの内積を求めます．この値が角度スペクトルになります．直交する二つのベクトルの内積は0となるため，スペクトルの値が0になる角度を電波の到来角度とみなせます．これは，鋭いヌルを利用することになり，高分解能の推定が可能になるわけです．これがメイン・ビームの操作によるビーム・フォーマ法とは大きく違う点です．

計算したスペクトルのスプリアスを抑えるために，一般には複数のノイズ固有ベクトルを用いて同じ処理を行い，スペクトルを平均化します．また，計算されたスペクトル値の逆数をとることで，**図5**のようにヌル点が鋭いピークとして現れることになります．**図5**のスペクトルは，4素子アレイ・アンテナを用いて三つの信号の到来方向を推定した結果となります．

$$P_{MU} = \frac{\boldsymbol{a}^{\mathrm{H}}(\theta) \cdot \boldsymbol{a}(\theta)}{\boldsymbol{a}^{H}(\theta) \cdot \boldsymbol{E}_N \boldsymbol{E}_N^{\mathrm{H}} \boldsymbol{a}(\theta)} \quad \cdots\cdots\cdots(7)$$

$\boldsymbol{E}_N = [\boldsymbol{e}_{L+1} \cdots \boldsymbol{e}_K]$：ノイズ固有ベクトル（$K$素子アレイに信号が$L$波到来する場合）

4．到来方向推定システム

ここで，横浜国立大学で試作された到来方向推定システムについて紹介します[(5), (7)]．推定手法は，複素数行列演算を実数のみの演算に低減できる行列変換（ユニタリ変換という）を用いたユニタリMUSIC法です．MUSIC法は，前述のように非常にシンプルな原理です．単純繰り返し処理に適しているため，FPGAを用いた高速演算が可能と考えられます．

● システムの構成

ディジタル信号処理回路は，大規模FPGA（米国Altera社のStratix「EP1S40」）に実装しました．システムの構成を**図6**に示します．ハードウェアは第2章で説明したものと同じです．到来方向推定における信号処理は，相関行列の計算，固有値分解処理，FFTによるスペクトル計算，スペクトル探索といった，大きく四つの処理プロセスに分けられます．

まず，8素子リニア・アレイ・アンテナから受信されたRF信号（5GHz）を，受信器で周波数変換し，A-Dコンバータで直接サンプリング可能な低いIF周波数（40MHz）にします．A-Dコンバータによりサンプリングし，12ビットの

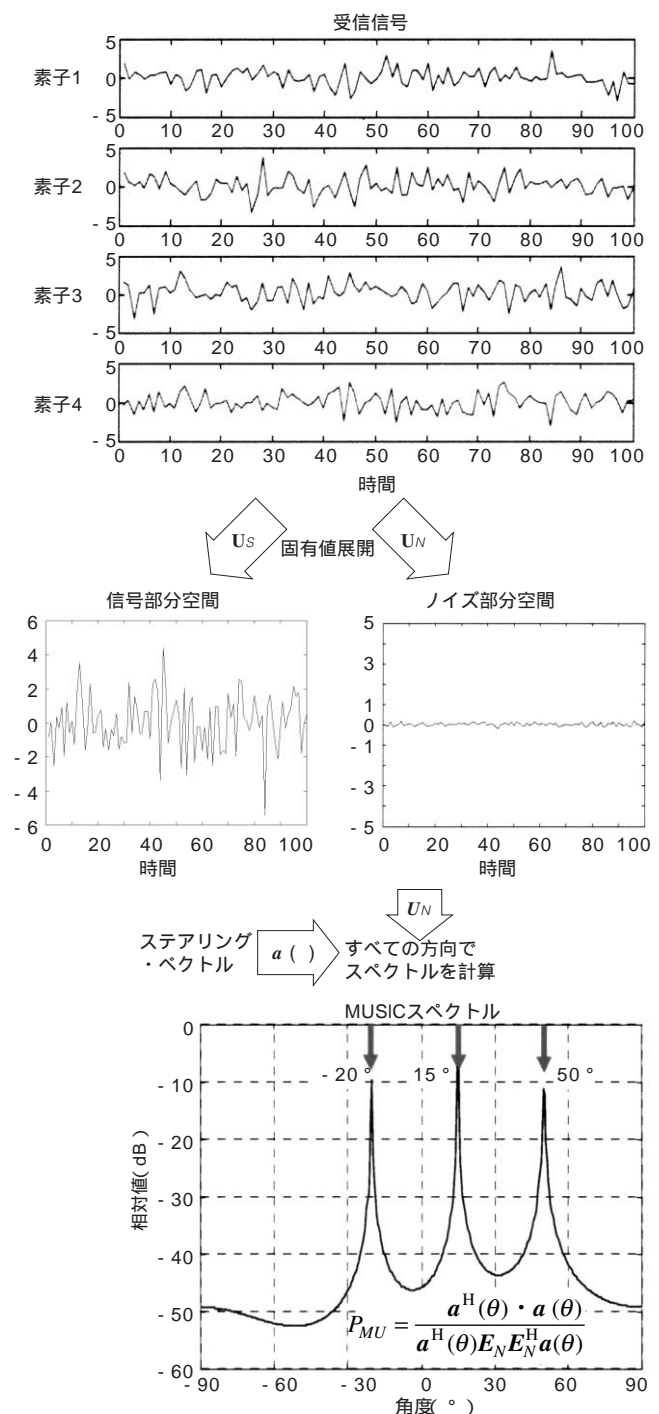



図5　MUSIC法の概念的な計算フロー

ディジタル・データとしてFPGAに入力します．この信号は，再びFPGAで複素ベースバンド信号に変換され，複素数の解析信号になります．

検波後の解析信号は一般にオーバサンプリングされているため，サンプル数を減らす必要があります．各素子からの信号をベクトルとして取り扱い，自己相関行列を作成します．ここで，内部処理は16ビットの固定小数点演算で行われます．それから，相関行列の固有値展開を行い，雑音固有ベクトルを計算します．

固有値展開は周期ヤコビ法（Cyclic Jacobi）を用いました[6]．これはベクトル回転の単純な繰り返しのみで実装できます．ベクトル回転動作を並列化することが容易なため，FPGA上の回路化にもっとも適しているように思われます．角度スペクトルの計算は，空間領域でのFFTを用いることにより高速処理が可能です．そのスペクトル値に対して1次元のピーク検索を行うことで，到来角を推定するという構成になります．

信号処理はA-D変換基板上のFPGAで行います．システム制御や外部インターフェースは，SH4 CPU上でNetBSDが動作する組み込みCPU基板を使用しました[28]．

● 信号処理のFPGA実装

ここで，今回設計した信号処理部のFPGAへの実装について簡略に説明します．誌面の関係で詳細な説明は省略してしまいますが，興味のある方は参考文献を参照してください．

ユニタリMUSIC法における信号処理の流れを**表1**に示します．第一ステップは，ユニタリ変換で入力信号を実数化することです．それから相関行列を作成し，平均処理を行います．8素子のアレイの場合に実数の相関行列は

$$R_{XX}=\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{18} \\ r_{21} & r_{22} & \cdots & r_{28} \\ \vdots & \vdots & \ddots & \vdots \\ r_{81} & r_{82} & \cdots & r_{88} \end{bmatrix} \quad \cdots\cdots(8)$$

のようになり，各要素は次の式で計算されます．

$$r_{pp}=\frac{1}{N}\sum_{n=0}^{N-1}\left\{x_{P,I}^{2}(n)+x_{P,Q}^{2}(n)\right\}$$
$$r_{pq}=\frac{1}{N}\sum_{n=0}^{N-1}\left\{\begin{array}{l}\left(x_{P,I}(n)x_{q,I}(n)-x_{p,Q}(n)_{q,Q}(n)\right)\\ +j\left(X_{p,Q}(n)x_{q,I}(n)+x_{p,I}(n)x_{q,Q}(n)\right)\end{array}\right\} \quad \cdots(9)$$

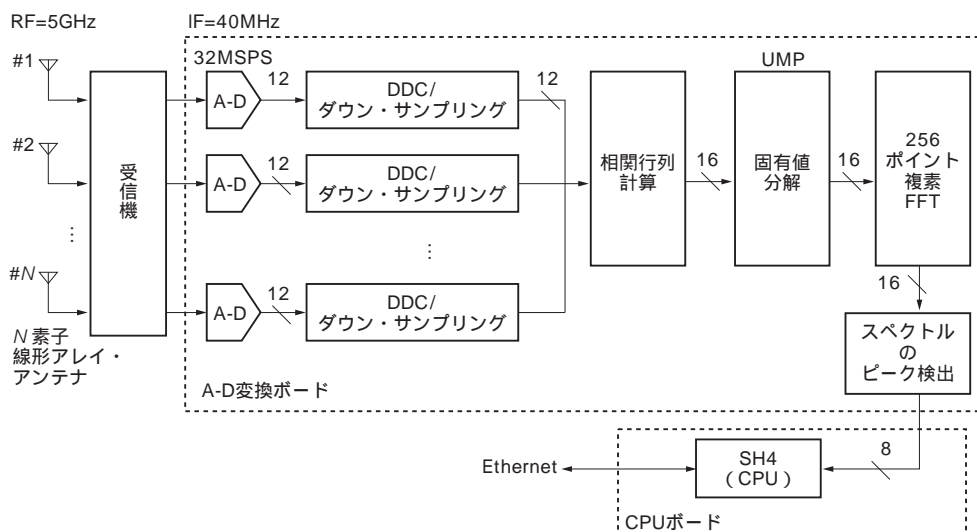



図6
FPGA上にUnitary MUSICアルゴリズムを実装したシステムのブロック図

表1
ユニタリMUSICにおける主な演算

演算式	処理内容
$Y_i=Q^{H}X_i$	ベクトルに対してユニタリ変換を行う
$R_{yy}(n)=\beta\cdot R_{yy}(n-1)+(1-\beta)\cdot\sum_{i=1}^{M}\mathrm{Re}\left[Y_i(n)Y_i^{H}(n)\right]$	空間平均処理と時間平均処理を行う
$E_i=EVD\left[R_{yy}\right]$	CORDICを用いるCyclic Jacobi固有値分解を行う
$\sum_{i=L+1}^{K}\left\|DFT\{Q\cdot E_i\}\right\|^2$	ノイズ固有ベクトルのユニタリ逆変換，FFTによる角度スペクトル計算

リスト1にVHDLによる記述を示します．64要素の中，対象性を用いて対角成分(r_{pp})の8要素と上三角成分(r_{pq})の28要素の乗算回路を並列で行うことになります．FPGAの専用乗算ブロックを用いることで，高速な回路実装が可能になります．作成された相関行列を用いて，固有値展開を行います．

ハードウェア実装の容易なCyclic Jacobi法を用いますが，回路はCORDIC法により固定小数点演算が実現できます[5]．固有値分解により得られたノイズ固有ベクトルを複素数に戻して，その離散フーリエ変換により角度スペクトルを計

リスト1　8素子のアレイ入力信号から相関行列を計算する回路（corrxx.vhd）

```
--************************************--
--      Compute Correlation Matrix
--          corrxx.vhd
--************************************--

-- Rxx = Real( Y*Hermitian(Y) )
-- 16 Snapshots
-- 12bit Inputs, 16bits Outputs

LIBRARY IEEE;
USE IEEE.std_logic_1164.all;
USE IEEE.std_logic_ARITH.all;
USE IEEE.std_logic_SIGNED.all;
USE work.util_package.ALL;

ENTITY corrxx IS
  PORT (
    clk   : IN std_logic;
    clkx  : IN std_logic;
    ena   : IN std_logic;
    reset : IN std_logic;

    i_xi  : IN slv12ar;  -- 12bits 4 array
    i_xq  : IN slv12ar;

    r     : OUT slv16ar36);
END ENTITY corrxx;

ARCHITECTURE rtl OF corrxx IS

  COMPONENT unitarytr IS
    PORT (
      clk   : IN std_logic;
      ena   : IN std_logic;
      reset : IN std_logic;

      xi  : IN slv12ar;  -- 12 bits 3 array
      xq  : IN slv12ar;

      yi  : OUT slv12ar;  -- 12 bits 3 array
      yq  : OUT slv12ar);
  END COMPONENT unitarytr;

  COMPONENT boxcar IS
    PORT (
      clk      : IN std_logic;
      reset    : IN std_logic;
      ena      : IN std_logic;
      x_in     : IN std_logic_vector(15 downto 0);
      y_out    : OUT std_logic_vector(15 downto 0));
  END COMPONENT boxcar;

  COMPONENT iirupdate IS
    PORT (
      clk      : IN std_logic;
      reset    : IN std_logic;
      ena      : IN std_logic;
      x_in     : IN std_logic_vector(15 downto 0);
      y_out    : OUT std_logic_vector(15 downto 0));
  END COMPONENT iirupdate;

  SIGNAL  ti1, tq1          : slv12ar;
  SIGNAL  yyi1, yyq1        : slv12ar;
  SIGNAL  yy1               : slv12ar36;
  SIGNAL  rr, in_r, reg_r   : slv16ar36;
  SIGNAL  mult_x, mult_y    : slv16ar36;
  SIGNAL  s                 : std_logic;
  BEGIN

    -- INPUTs
    ti1(1 TO 8) <= i_xi(1 TO 8);
    tq1(1 TO 8) <= i_xq(1 TO 8);

    unitarytr1 : unitarytr PORT MAP (clk, '1', reset,
ti1, tq1, yyi1, yyq1);


loop_r_jj:    FOR j IN 1 TO 8 GENERATE
      rr(j   )<=
fixadd(fixmult(yyi1(j),yyi1(j),16),
                   fixmult(yyq1(j),yyq1(j),16)); -- r_jj
    END GENERATE;
loop_r_1j:    FOR j IN 2 TO 8 GENERATE
      rr(j+ 7)<=
fixadd(fixmult(yyi1(1),yyi1(j),16),
                   fixmult(yyq1(1),yyq1(j),16)); -- r_1j
    END GENERATE;
loop_r_2j:    FOR j IN 3 TO 8 GENERATE
      rr(j+13)<=
fixadd(fixmult(yyi1(2),yyi1(j),16),
                   fixmult(yyq1(2),yyq1(j),16)); -- r_2j
    END GENERATE;
loop_r_3j:    FOR j IN 4 TO 8 GENERATE
      rr(j+18)<=
fixadd(fixmult(yyi1(3),yyi1(j),16),
                   fixmult(yyq1(3),yyq1(j),16)); -- r_3j
    END GENERATE;
loop_r_4j:    FOR j IN 5 TO 8 GENERATE
      rr(j+22)<=
fixadd(fixmult(yyi1(4),yyi1(j),16),
                   fixmult(yyq1(4),yyq1(j),16)); -- r_4j
    END GENERATE;
loop_r_5j:    FOR j IN 6 TO 8 GENERATE
      rr(j+25)<=
fixadd(fixmult(yyi1(5),yyi1(j),16),
                   fixmult(yyq1(5),yyq1(j),16)); -- r_5j
    END GENERATE;
loop_r_6j:    FOR j IN 7 TO 8 GENERATE
      rr(j+27)<=
fixadd(fixmult(yyi1(6),yyi1(j),16),
                   fixmult(yyq1(6),yyq1(j),16)); -- r_6j
    END GENERATE;
      rr(36)  <=
fixadd(fixmult(yyi1(7),yyi1(8),16),
                   fixmult(yyq1(7),yyq1(8),16)); -- r_78

loop0: FOR j IN 1 TO 36 GENERATE
    PROCESS (clk, ena, reset) IS
      BEGIN
        IF reset = '1' THEN
          in_r(j)      <= (OTHERS=>'0');
        ELSIF clk'event AND clk = '1' THEN
          in_r(j)      <= rr(j);
        END IF;
      END PROCESS;
    END GENERATE; -- end loop0

lpf_loop0 : FOR i IN 1 TO 36 GENERATE
      lpfuu : iirupdate PORT MAP
          (clk, reset, ena, in_r(i), r(i));
    END GENERATE lpf_loop0;


  END ARCHITECTURE rtl;
```

表2　ユニタリMUSICのFPGA実装結果（準同期検波処理を含む）

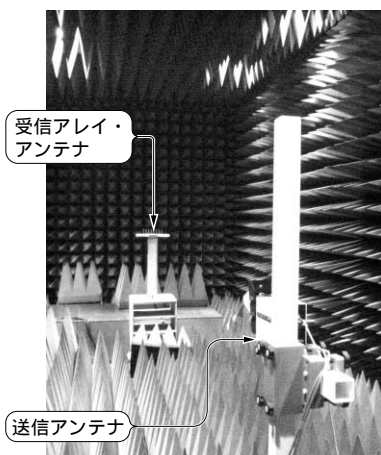
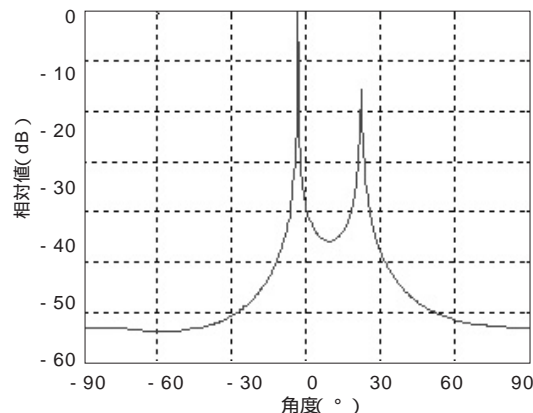
アンテナ素子数	使用LE数
4素子（空間平均処理付き）	12,007
4素子	12,995
8素子	29,472

（a）電波暗室における実験の様子

（b）8素子アレイ・アンテナに2波がそれぞれ−5°と20°から到来する場合の実験結果

図7　ユニタリMUSIC法で計算した角度スペクトルの実践

算します．フーリエ変換はFPGAベンダ提供のIP（Intellectual Property）コアなどを用いれば手軽に実装できます．この角度スペクトルからピーク点を検索することで，到来方向が推定されます．

表2にAltera社のStratix（EP1S40）を用いて実装した結果を示します．並列処理をすることを考慮し実装面積はやや大きめとなりましたが，8チャネルの準同期検波を含めて8素子アレイ・アンテナの実装まで1チップに収まりました．

図7（a）は吸収体により反射が抑えられる電波暗室における実験の様子を示します．8素子アレイ・アンテナに−5°と20°からそれぞれ2波が到来する場合の結果を**図7（b）**に示します．到来方向が正しく推定されることが分かります．

● 終わりに

今回の特集は，最近の移動通信の動向やMIMOなどで注目を集めているアレイ・アンテナ信号処理技術の基礎原理について専門外の人にも優しく解説し，FPGAを用いた実装例も紹介しました．少し専門的な話になってしまったところもありましたが，興味のある方はぜひ，参考文献を読んで頂きたいと思います．

これまで1対1で扱われた情報が，複数のストリームで伝送され，信号処理により合成されるようになりました．より高速かつ並列な処理が必要になってくる中，FPGA特有の信号処理能力はとても魅力的であると思います．無線通信分野におけるFPGAの活躍を期待したいと思います．

参考・引用*文献

(1) 辻 宏之；アレーアンテナを用いた屋内外の無線局位置推定の実験的検証，電子情報通信学会論文誌，vol. J90-B，no. 9，pp.784-796.
(2) 菊間信良；アダプティブアンテナ技術，2003年，オーム社.
(3) R. Schmidt；“*Multiple emitter location and signal parameter estimation,*” IEEE Transactions on Antennas and Propagations, vol. 34, no. 3, pp. 276-280, March 1986.
(4) R. Roy and T. Kailath；“*ESPRIT - Estimation of signal parameters via rotational invariance techniques,*”IEEE Transactions on Acoustics , Speech, and Signal Processing, vol. ASSP-37, pp. 984-995, 1989.
(5) Minseok Kim, Koichi Ichige and Hiroyuki Arai；“*Implementation of FPGA based Fast Unitary MUSIC DOA Estimator*,” IEICE Transactions on Electronics, Vol. E87-C, No. 9, pp.1485-1494, Sept. 2004.
(6) Minseok Kim, Koichi Ichige and Hiroyuki Arai；“*Design of Jacobi EVD processor based on CORDIC for DOA estimation with MUSIC algorithm,*” IEICE Transactions on Communications, Vol. E85-B, No.12, pp. 2648-2655, Dec. 2002.
(7) Minseok Kim, Koichi Ichige and Hiroyuki Arai；“*16-element DOA estimation system,*” IEICE Technical Report, SR2005-43, YRP, Japan, July 2005.
(8) CPU board，ブレインズ，http://www.brains.co.jp/

Minseok Kim
東京工業大学